workshop

# Blazorfy

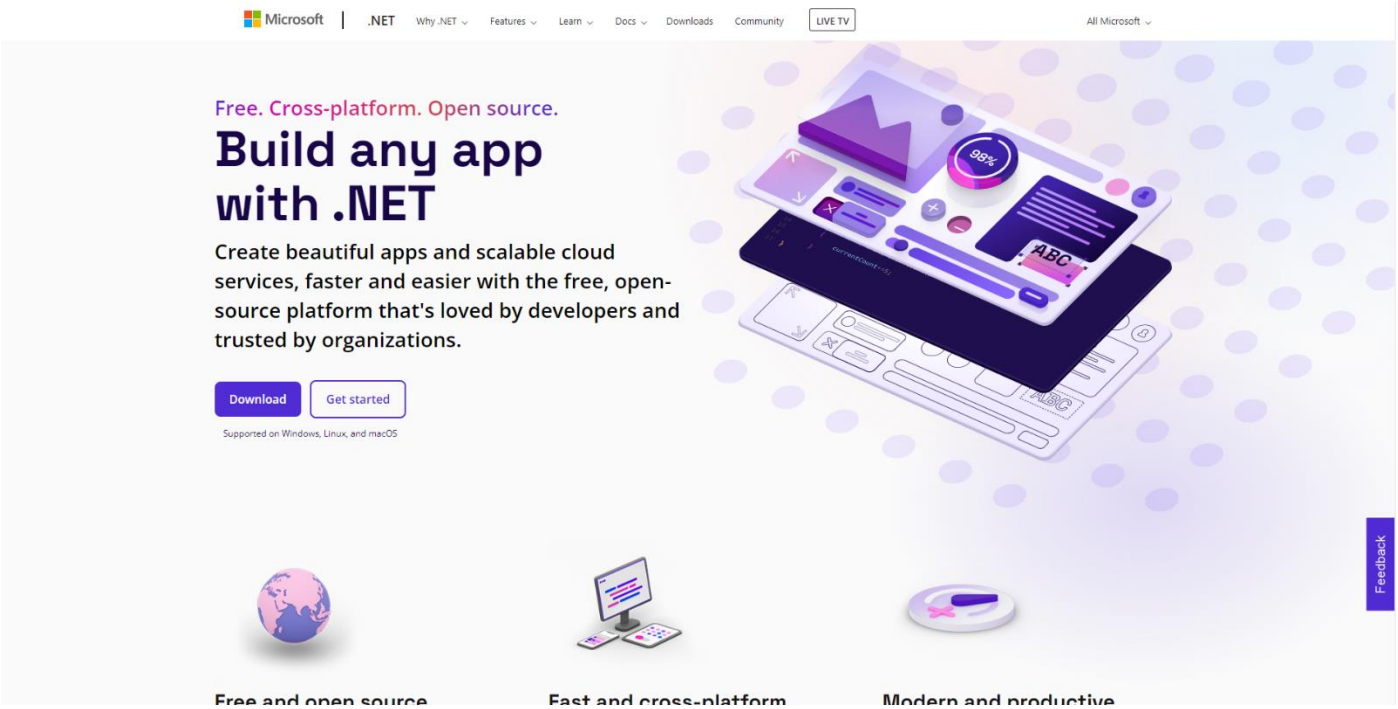
# Setup



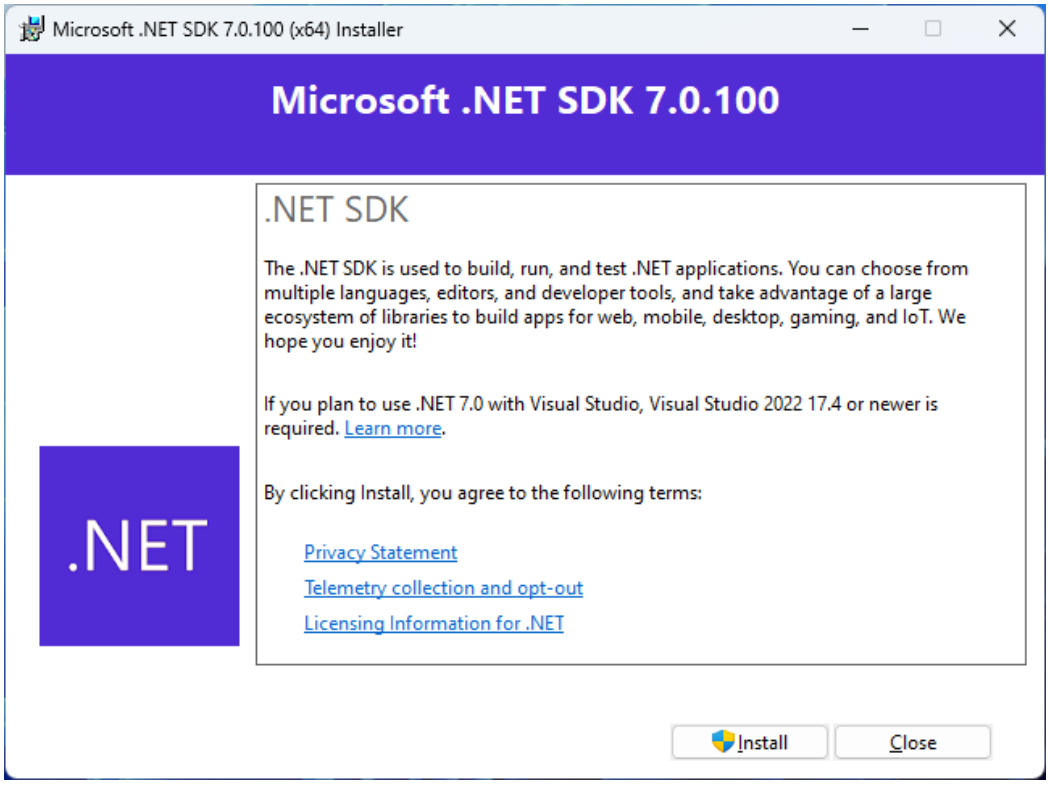
# Setup

## .NET

**.NET** includes **Blazor** so you will need to **Download** and **Install** the latest version of the **.NET SDK**, which if you don't have it already you can **Download** it for **Windows** or **Mac** from [dot.net](dot.net)

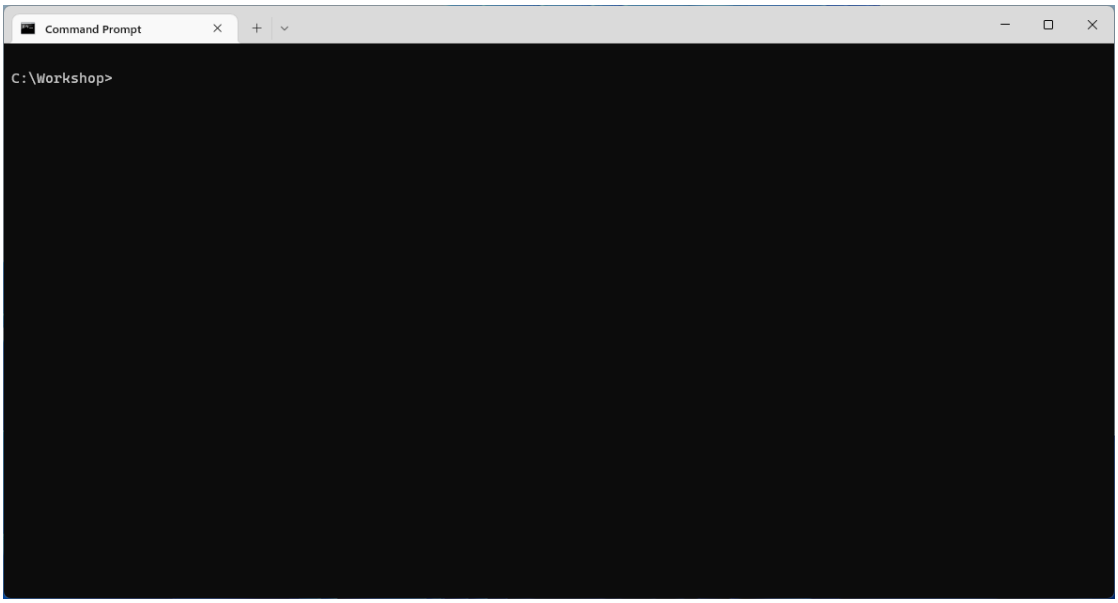



Once **Downloaded** you can **Install** the **.NET SDK** by following the steps in the **Installation Wizard**



## Project

If the **.NET SDK** has been **Installed**, then if using a **Mac** you then need to go to **Finder** then search for **Terminal** and then select it or if using **Windows** you need to go to **Start** then search for **Command Prompt** and then select it so it launches as follows:




Once in the **Command Prompt** or **Terminal** you will need to create a new **Folder**, you can use **mkdir** followed by the name of the **Folder** e.g. *Workshop* and then press **Enter**.

```
mkdir Workshop
```

Then you will need to switch to this **Folder**, to do this from the **Command Prompt** or **Terminal** type in the following command and then press **Enter**:

```
cd Workshop
```

Once in this **Folder** you can create a new **Project** using the **.NET CLI** that was **Installed** as part of the **.NET SDK**. While still in the **Command Prompt** or **Terminal** type in the following and then press **Enter**:

```
dotnet new blazorwasm -o Blazorfy
```

This will create a new **Project** for **Blazor** using **WebAssembly** or **wasm**. Once the **Project** has been created in the **Command Prompt** or **Terminal** you will need to change to the **Folder** for the **Workshop** by typing in the following and then press **Enter**:

```
cd Blazorfy
```

## Packages

While still in the **Command Prompt** or **Terminal** you will add some **Packages** that will be used in **Blazorfy** to add the first **Package** of *Blazored.LocalStorage*, type the following and then press **Enter**:

```
dotnet add package Blazored.LocalStorage
```

This will add the **Package** for *Blazored.LocalStorage* created by *Chris Sainty* which provides access to local storage for **Blazor** applications, this will be used to save and load values in the **Browser**.

Then while still in the **Command Prompt** or **Terminal** you can add the second **Package** of *Spotify.NetStandard* type the following and then press **Enter**:

```
dotnet add package Spotify.NetStandard
```

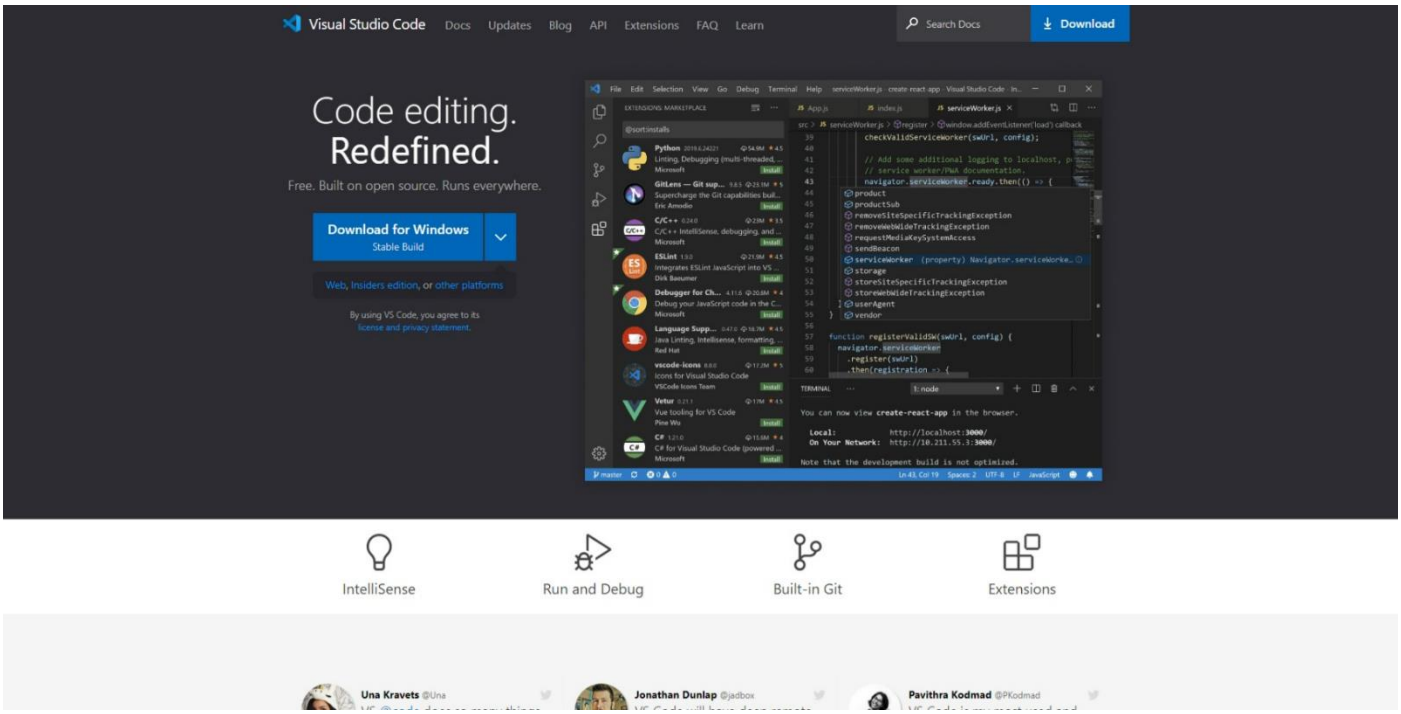
This will add the **Package** for *Spotify.NetStandard* created by *Peter Bull* which provides access to the **Spotify Web API** and will be used to obtain information from **Spotify**.

You can then close this **Command Prompt** or **Terminal** as it will no longer be needed in the **Workshop**.

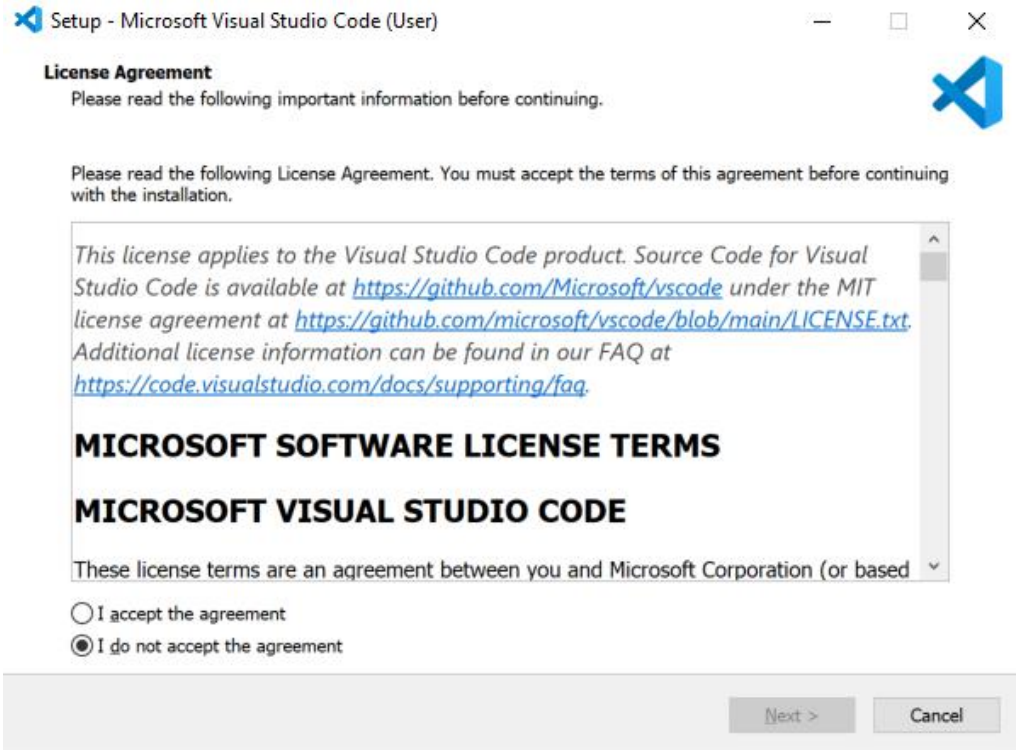
**Packages** provide functionality to developers written by other developers for .**NET**. You can find out details of other **Packages** that are available at nuget.org.

## Visual Studio Code

**Visual Studio Code** is a free **Integrated Development Environment or IDE** created by **Microsoft** and will be used in the **Workshop** and will make writing the application easier. You can **Download** it, if you don't have it already, for **Windows** or **Mac** from [code.visualstudio.com](code.visualstudio.com)




Once it has been **Downloaded,** you can then **Install** it by following the steps in the **Installation Wizard**




Once you've installed **.NET**, used **`dotnet new blazorwasm -o Blazorfy`**, added the **Packages** and installed **Visual Studio Code** then you're ready for the **Workshop**.